IN THE CLAIM

1. (Previously Amended, Currently Amended) A method for managing a memory system having a plurality of subsystems, comprising the steps of:

upon accessing the subsystems for a piece of data used by a first process,

determining an access time to acquire the piece of data in the memory system;

comparing the determined access time to a threshold; and

taking an action based on results of the comparing step;

wherein

~~a value of the threshold is selected based on whether the value is a realistic time for a memory access;~~

a memory table includes entries pointing to data blocks storing data for at least one subsystem;

the entries are used to locate the data stored in the data blocks; and

while the first process is being executed, the memory table working with a memory manager managing the data blocks independent of an operating system working with the memory system and independent of a processor working with the memory system.

2. (Previously Amended) The method of claim 1 wherein a data block containing the piece of data is placed in the memory system based on information selected in one or a combination of:

a movement pattern of data in the data block,

a structure of the memory system, and

a cache-level architecture in the memory system.

3. (Canceled)

4. (Canceled)

5. (Previously Amended) The method of claim 1 further comprising the steps of:

the memory table using a physical address of a memory page corresponding to the piece of data to convert to a location address corresponding to an entry pointing to the location of the piece of data.

6. (Previously Amended, Currently Amended) A method for managing a memory system, comprising the steps of:

upon accessing the memory system for a piece of data used by a first process,

a processor working with the memory system continuing its functions until it is stalled;

comparing a time taken to complete the memory access to a threshold; and

taking an action based on results of the comparing step~~; a value of the threshold being selected based on whether the value is a realistic time for a memory access~~.

7. (Original) The method of claim 6 wherein the action is selected in one or a combination of

postponing executing the first process and allowing executing a second process;

causing the first process to be switched to a second process; and

causing a performance monitor on the memory system or on a system using the memory subsystem.

8. (Original) The method of claim 6 further comprising the step of polling a latency manager for the time taken to complete the memory access; the latency manger being part of managing the memory system.

9. (Previously Amended) The method of claim 6 further comprising the steps of:

using a memory table having entries pointing to data blocks storing data for at least one subsystem; and

using the entries to locate the data stored in the data blocks.

10. (Previously Amended) The method of claim 9 wherein, while the first process is being executed, the memory table working with a memory manager managing the data blocks independent of a processor working with the memory system and independent of an operating system working with the memory system.

11. (Previously Amended, Currently Amended) A method for managing a memory system, comprising the steps of:

upon accessing the memory system for a piece of data used by a first process

counting a time elapsed from the time the data access starts; the counted time being increased as the data is being accessed;

comparing the counted time to a threshold; ~~a value of the threshold is selected based on whether the value is a realistic time for a memory access;~~ and

based on results of the comparing step, taking an action selected in one or a combination of

postponing executing the first process and allowing executing a second process;

causing the first process to be switched to a second process; and

causing a performance monitor on the memory system or on a system using the memory system.

12. (Previously Amended) The method of claim 11 further comprising the steps of:

using a memory table having entries pointing to data blocks storing data for at least one memory subsystem; and

using the entries to locate the data stored in the data blocks.

13. (Previously Amended, Currently Amended) A computer-readable medium embodying instructions for a computer to perform a method for managing a memory system having a plurality of subsystems, the method comprising the steps of:

upon accessing the subsystems for a piece of data used by a first process, determining an access time to acquire the piece of data in the memory system;

comparing the determined access time to a threshold; and

taking an action based on results of the comparing step;

wherein

~~a value of the threshold is selected based on whether the value is a realistic time for a memory access;~~

a memory table includes entries pointing to data blocks storing data for at least one subsystem;

the entries are used to locate the data stored in the data blocks; and

while the first process is being executed, the memory table working with a memory manager managing the data blocks independent of an operating system working with the memory system and independent of a processor working with the memory system.

14. (Previously Amended) The computer-readable medium of claim 13 wherein a data block containing the piece of data is placed in the memory system based on information selected in one or a combination of:

a movement pattern of data in the data block,

a structure of the memory system, and

a cache-level architecture in the memory system.

15. (Canceled)

16. (Canceled)

17. (Previously Amended) The computer-readable medium of claim 13 wherein the method further comprises the steps of:

the memory table using a physical address of a memory page corresponding to the piece of data to convert to a location address corresponding to an entry pointing to the location of the piece of data.

18. (Previously Amended, Currently Amended) A computer-readable medium embodying instructions for a computer to perform a method for managing a memory system, the method comprising the steps of:

upon accessing the memory system for a piece of data used by a first process,

a processor working with the memory system continuing its functions until it is stalled;

comparing a time taken to complete the memory access to a threshold; ~~a value of the threshold being selected based on whether the value is a realistic time for a memory access;~~ and

based on results of the comparing step, taking an action.

19. (Original) The computer-readable medium of claim 18 wherein the method further comprises the step of polling a latency manager for the time taken to complete the memory access; the latency manger being part of managing the memory system.

20. (Previously Amended) The computer-readable medium of claim 18 wherein the method further comprises the steps of:

using a memory table having entries pointing to data blocks storing data for at least one subsystem; and

using the entries to locate the data stored in the data blocks.

21. (Previously Amended, Currently Amended) A computer-readable medium embodying instructions for a computer to perform a method for managing a memory system, the method comprising the steps of:

upon accessing the memory system for a piece of data used by a first process,

counting a time elapsed from the time the data access starts; the counted time being increased as the data is being accessed;

comparing the counted time to a threshold~~, a value of the threshold being selected based on whether the value is a realistic time for a memory access~~; and

based on results of the comparing step, taking an action selected in one or a combination of

postponing executing the first process and allowing executing a second process;

causing the first process to be switched to a second process; and

causing a performance monitor on the memory system or on a system using the memory subsystem.

22. (Previously Amended) The computer-readable medium of claim 21 wherein the method further comprises the steps of:

using a memory table having entries pointing to data blocks storing data for at least one memory subsystem; and

using the entries to locate the data stored in the data blcoks.

23. (Previously Amended, Currently Amended) An apparatus for managing a memory system having a plurality of subsystems, comprising:

means for, upon accessing the subsystems for a piece of data used by a first process,

determining an access time to acquire the piece of data in the memory system;

comparing the determined access time to a threshold; and

taking an action based on results of the comparing step;

wherein

~~a value of the threshold is selected based on whether the value is a realistic time for a memory access;~~

a memory table includes entries pointing to data blocks storing data for at least one subsystem;

the entries are used to locate the data stored in the data blocks; and

while the first process is being executed, the memory table working with a memory manager managing the data blocks independent of an operating system working with the memory system and independent of a processor working with the memory system.

24. (Previously Amended) The apparatus of claim 23 wherein a data block containing the piece of data is placed in the memory system based on information selected in one or a combination of:

a movement pattern of data in the data block,

a structure of the memory system, and

a cache-level architecture in the memory system.

25. (Canceled)

26. (Canceled)

27. (Previously Amended) The apparatus of claim 23 wherein the memory table using a physical address of a memory page corresponding to the piece of data to convert to a location address corresponding to an entry pointing to the location of the piece of data.

28. (Previously Amended, Currently Amended) An apparatus for managing a memory system, comprising:

upon accessing the memory system for a piece of data used by a first process,

a processor for working with the memory system and for continuing its functions until it is stalled;

means for comparing a time taken to complete the memory access to a threshold; ~~a value of the threshold being selected based on whether the value is a realistic time for a memory access;~~ and

means for taking an action based on results of comparing.

29. (Original) The apparatus of claim 28 further comprising means for polling a latency manager for the time taken to complete the memory access; the latency manger being part of managing the memory system.

30. (Previously Amended) The apparatus of claim 28 further comprising a memory table having entries pointing to data blocks storing data for at least one subsystem; the entries being used to locate the data stored in the data blocks.

31. (Previously Amended, Currently Amended) An apparatus for managing a memory system, comprising:

upon accessing the memory system for a piece of data used by a first process,

means for counting a time elapsed from the time the data access starts; the counted time being increased as the data is being accessed;

means for comparing the counted time to a threshold~~, a value of the threshold being selected based on whether the value is a realistic time for a memory access~~; and

means for taking an action selected in one or a combination of postponing executing the first process and allowing executing a second process;

causing the first process to be switched to a second process; and

causing a performance monitor on the memory system or on a system using the memory subsystem.

32. (Previously Amended) The apparatus of claim 31 further comprising a memory table having entries pointing to data blocks storing data for at least one memory subsystem; the entries being used to locate the data stored in the data blocks.

33. (Previously Presented) The method of claim 5 wherein the physical address of the memory page is converted from a virtual address of the piece of data.

34. (Previously Presented) The computer-readable medium of claim 17 wherein the physical address of the memory page is converted from a virtual address of the piece of data.

35. (Previously Presented) The apparatus of claim 27 wherein the physical address of the memory page is converted from a virtual address of the piece of data.